caption

Aalborg University
Semester Project
Signal Processing and Computing
group 770

Packet Loss Concealment for Digital TV via Convex Optimization

participants:
Mohammad El-Sayed
Søren Lund
Konstantinos Voulgaris

supervisors:
Jan Østergaard
Adel Zahedi

AALBORG UNIVERSITET

STUDENTERRAPPORT

Department of Electronic systems
Fredrik Bajers Vej 7
9220 Aalborg E
Phone: 96 35 86 90
http://es.aau.dk

Title: Packet Loss Concealment for Digital TV by Sparse DCT Approximation

Theme: Signal Processing and Computing

Project Period: Sep. 2nd - Dec. 17th 2014

Project Group: 14gr770

Participants:

Mohammad El-Sayed
Søren Lund
Konstantinos Voulgaris

Supervisors:

Jan Østergaard
Adel Zahedi

Number of Copies: 2
Pages: 39
Number of Appendixes: 1
Finished 17th of December 2014

Mohammad El-Sayed

Søren Lund

Konstantinos Voulgaris

Preface

This study is produced and written by 1st semester students from Signal Processing and Computing from group 770 at Aalborg University with Jan Østergaard and Adel Zahedi as supervisors. The paper is split up into worksheets each marked with a unique number, the purpose of the worksheets is to document different subjects which have been investigated during the project period. Based on our work a scientific article is written, where the main contributions to the field of image processing are documented. This study along the with a poster, are presented at the SEMCON conference 2014.

- All figures and tables are provided with a caption and a unique number according to the worksheet.
- The respective number and caption are placed after the figure or table.
- All equations are referred to by number, but with no caption.
- Citations are presented by indices in square brackets, i.e. [*index*]. The bibliography is displayed after the last worksheet.
- All MATLAB scripts can be found in the attached CD.

Abbreviations

CAD	Content Adaptive Technique
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
MSE	Mean Square Error
PLC	Packet Loss Concealment
PSNR	Peak Signal to Noise Ratio
OAI	Orientation Adaptive Interpolation
SDA	Sparse DCT Approximation
SDA-O	Sparse DCT Approximation Oracle
SDA-PS	Sparse DCT Approximation using a Patch Search
SDA-PS-A	SDA Patch Search using an Average approach
SDA-PS-M	SDA Patch Search using a Median approach
SDA-PS-RW	Sparse DCT Approximation with Reweighted ℓ_1-norm
SDA-3	Sparse DCT Approximation with a frame size of 3 pixels
SSIM	Structural Similarity Index Measurement

Contents

1 Definitions

1.1 Packet Loss Concealment

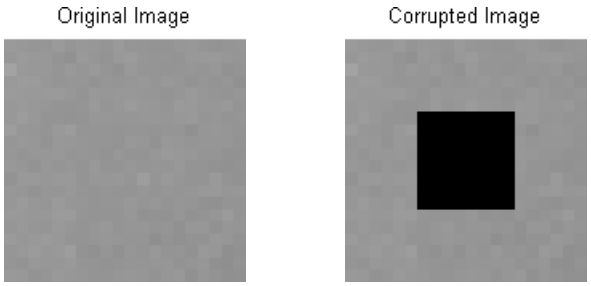
When transmitting digital TV over error prone channels, i.e. both wired and wireless, errors occasionally occur. In digital TV, signals are encoded in bit-streams, each one containing an image-package of a certain size. The bit-stream is later decoded at the receiver end, however if packages are corrupted during transmission they can not be decoded correctly at the receiver, and visual artifacts will appear in the image. Packet loss concealment (PLC) is a technique designed to minimize the effect of lost packets, exploiting a subset of pixels in a neighborhood of the lost block. Mostly all coding standards such as JPEG and H.264, processes image data in blocks of 8×8 pixels, with some standards supporting several block sizes[1][2]. Throughout this paper, concealment will therefore be done for image blocks of size 8×8 pixels. Furthermore the images used for simulations, suffer from a predefined amount of lost blocks, an example of a corrupted image is shown in Figure 1.1(b).

(a) Original Image (b) Corrupted Image

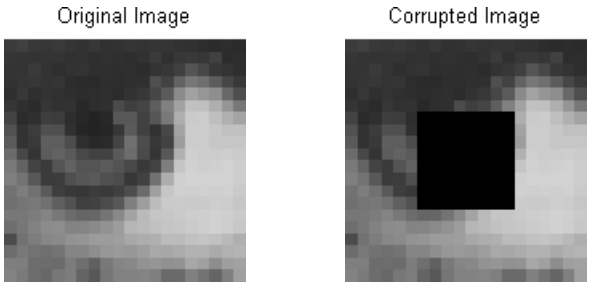
Figure 1.1.

Initial tests are performed for each PLC method throughout the paper, in order to have an intuition of the strengths and weaknesses of the methods. Specific locations for two error blocks are chosen in advance, one is located in a subpart of the image with only

small variations, i.e. a low frequency block, and the other block is located in a highly detailed area, i.e. a high frequency block. More specifically the low frequency block is located in the background of the *Lena* image, whereas the high frequency block is placed in the eye of *Lena*, a zoomed out version of both errors are shown in Figure 1.2. The initial tests provides a basis for a comparison between different PLC techniques, both by visual inspection and objective measures.



(a) Low frequency block (b) Corrupted low frequency block



(c) High frequency block (d) Corrupted high frequency block

Figure 1.2. Standard error block locations used in initial tests

A more comprehensive test for evaluating the performance of the PLC methods is conducted, it makes use of four standard images: *Lena*, *Barbara*, *Cameraman* and *Boat*, which are shown in Figure 1.3. Several images with randomly introduced errors, are used in order to give an average performance of some chosen PLC methods. All images are tested in grayscale with a size of 512×512 pixels and a bit-depth of 8, which corresponds to a pixel intensity range from 0 to 255.

(a) Lena (b) Barbara (c) Boat (d) Cameraman

Figure 1.3. Standard test images

2 Discrete Cosine Transform

There is a need for a transform, that can take an image from the spatial domain to the frequency domain. A technique widely used for image and video applications, is denoted as the discrete cosine transform (DCT). The name is derived from the fact that the basis functions of the transform, are defined as cosine terms. The transform has only real coefficients, and is defined as seen in Equation (2.1), for the 1-dimensional case [3].

$$C(k) = \alpha(k) \sum_{n=0}^{N-1} f(n) \cdot \cos\left[\frac{\pi(2n+1)k}{2N}\right] \quad \text{for} \quad k = 0, 1, 2, \ \ldots \ , N-1 \tag{2.1}$$

where

$$\alpha(k) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for} \quad k = 0 \\ \sqrt{\frac{1}{2N}} & \text{for} \quad k \neq 0 \end{cases} \tag{2.2}$$

The basic idea behind the DCT is to decompose the input, $f(n)$, into a linear combination of the basis functions. That is $f(n)$ given as a weighted sum of the cosine basis functions as seen in (2.3).

$$f(n) = a_1\vec{v}_1 + a_2\vec{v}_2 + \ldots + a_N\vec{v}_N \tag{2.3}$$

where

$\vec{v}_i \in \mathbb{R}^N$ is the i^{th} cosine basis function written on vector form
$a_i \in \mathbb{R}^1$ is the weighting coefficient belonging to the i^{th} basis function

Hence the DCT gives an idea on how much each basis function contributes to $f(n)$. All cosine basis functions are orthogonal, thus a multiplication of any two basis functions with different k values will yield a zero, or equivalently the dot product is zero. Whereas the dot product of any basis function by itself yields a constant scalar value.

$$v_i^\intercal v_j = \begin{cases} c & \text{for} \quad i = j \\ 0 & \text{for} \quad i \neq j \end{cases} \tag{2.4}$$

where

$c \in \mathbb{R}^1$ is a constant value

Furthermore all basis functions are linearly independent, meaning no basis function can be produced as a combination of the others.

It is clear from equation (2.1) that the cosine terms are being computed for increasing k values, hence increasing the frequency of the basis functions. The first coefficient given for $k = 0$, represents the amplitude at zero frequency and is denoted as the DC-coefficient. The coefficient $C(0)$ will return the scaled mean value of the sample sequence $f(n)$. All remaining coefficients for k, are referred to as AC coefficients. A discrete plot is presented in Figure 2.1, depicting all basis functions with a length N equal to 8.

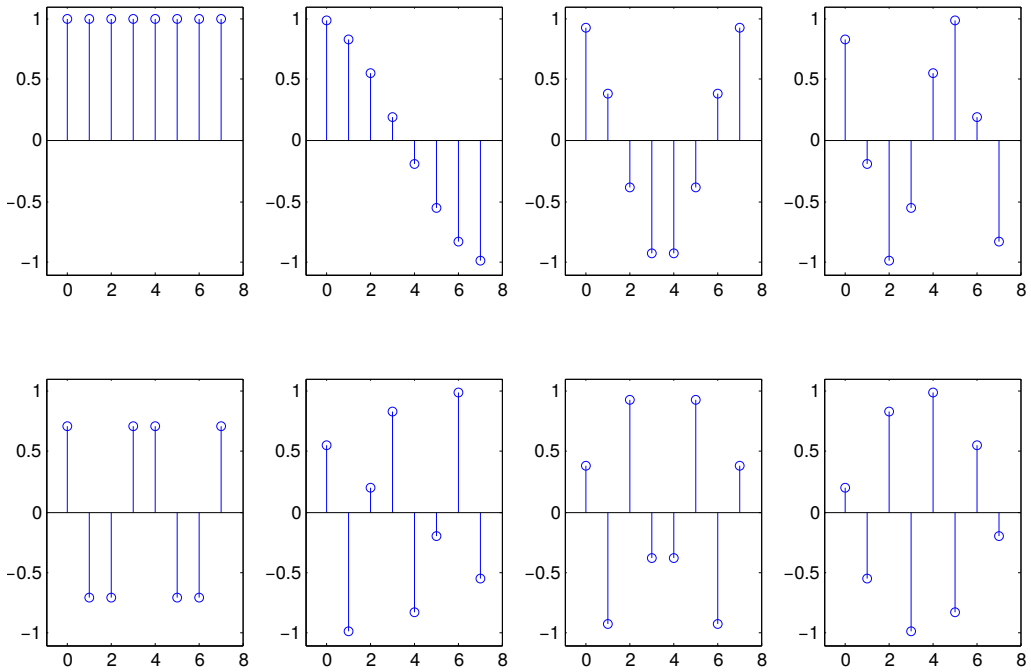


Figure 2.1. Discrete plot of the DCT basis functions for $N = 8$ and $k = 0, 1, \ldots, 7$.

In accordance with previous observations, the upper leftmost sequence in Figure 2.1, renders a DC signal. The remaining sequences are waveforms that increases progressively in frequency as k is incremented.

Since natural images typically are sparse in the frequency domain, their DCT representations tend to have more of its energy concentrated in a few number of coefficients. Consequently the original signal can be reconstructed from the inverse DCT (IDCT), by utilizing only a small set of DCT coefficients. The 1-dimensional IDCT is defined in equation (2.5).

$$f(n) = \sum_{k=0}^{N-1} \alpha\,(k) \cdot C(k) \cdot \cos\left[\frac{\pi(2n+1)k}{2N}\right] \quad \text{for} \quad k = 0, 1, 2, \ \ldots\ , N-1 \tag{2.5}$$

where

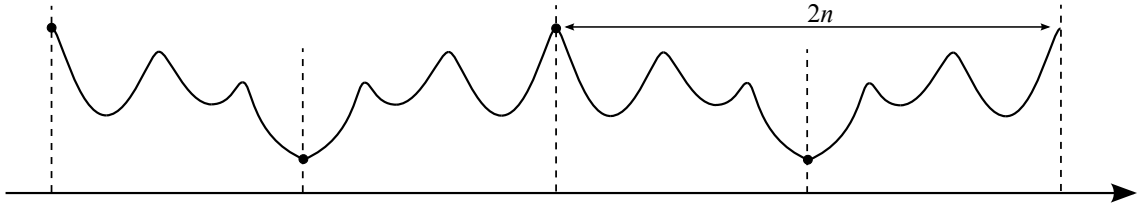
$\alpha(k)$ is defined in equation (2.2).

Because of the energy compaction property of the DCT, it is widely used in lossy image compression. Compression techniques such as JPEG, utilizes that only coefficients containing a significant amount of energy have a visible impact of the restored image, then by excluding low energy coefficients, the data storage is simultaneously reduced.

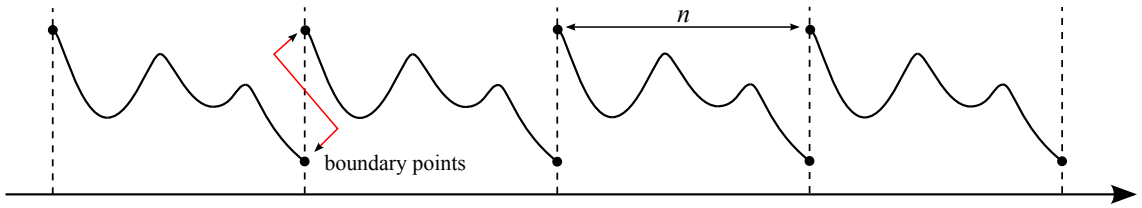
2.1 Boundary Properties

In particular, a rule that is applied for frequency transforms such as the DCT and as well the DFT, tells that any discontinuities introduced to a signal, will require more sinusoid terms in order to represent the signal. The smoother the function is, the fewer terms are required to represent it accurately, and the frequency representation will become more sparse. However the DCT and DFT are different with respect to the assumption about periodicity. The DCT has a implicit $2n$-point assumption that there exist even symmetry about the border points, since the cosine function is an even function. Meaning that there will be a continuous extension of the signal, at the boundaries, as shown in Figure 2.2(a)[4].

Whereas the DFT, assumes a input signal $f(n)$ to be periodic, meaning that the signal will repeat itself at the boundaries, without the transition necessarily being continuous. Discontinuities at the boundary will result in high frequency transform content. The periodicity assumption is depicted in Figure 2.2(b). For these reasons the DCT generally performs better in terms of energy compaction.



(a) Depicting the periodicity assumption for the DCT [4]



(b) Depicting the periodicity assumption for the DFT [4].

Figure 2.2.

Relating the respective boundary assumption for the two types of transforms to images, it becomes obvious that the DCT is more applicable in terms of returning a sparse representation. The DFT boundary assumption, expects from all the pixels in a $n \times n$ block, to be similarly repeated for all remaining blocks. It implies that any pixel value in any two image blocks with the same row and column index i and j, must be equal. Since both a vertical and horizontal transform is needed for an image matrix, the assumption must equally hold for the 2-dimensional case.

The DCT periodicity assumption is generally more reasonable, since it only requires two neighboring blocks to be a mirrored version of one another.

2.2 2D DCT

When applying DCT on an image represented by an 2D-array, it must necessarily be done in both the dimensions. Therefore equation (2.1) is redefined to include a transformation for all rows and columns[3]:

$$C(k,l) = \alpha\left(k\right)\beta\left(l\right)\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} f(m,n)\cos\left[\frac{\pi(2n+1)k}{2N}\right]\cos\left[\frac{\pi(2m+1)l}{2M}\right] \tag{2.6}$$

where

$$k = 0, 1, \ \ldots\ , N-1 \ \ \wedge \ \ l = 0, 1, \ \ldots\ , M-1 \tag{2.7}$$

and

$$\alpha(k) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for} \quad k = 0 \\ \sqrt{\frac{1}{2N}} & \text{for} \quad k \neq 0 \end{cases} \tag{2.8}$$

and

$$\beta(l) = \begin{cases} \sqrt{\frac{1}{M}} & \text{for} \quad l = 0 \\ \sqrt{\frac{1}{2M}} & \text{for} \quad l \neq 0 \end{cases} \tag{2.9}$$

Equation (2.6) generates a 2D DCT matrix C with dimensions $m \times n$. The case when both k and l are zero, is referred to as the DC coefficient, and is located in the upper left corner of C. Since the frequency transformation is done in two dimensions, it is essential to distinguish between horizontal and vertical frequency components. From the DC coefficient to the rightwards direction an increment in the horizontal frequency components appear, whereas the vertical frequencies increment downwards.

For each instance of k and l, a basis function is generated with dimensions $m \times n$, opposed to a vector as earlier. As a consequence there will be generated a total number of $M \cdot N$ basis functions, each one represented by a matrix. When k is incremented, the basis function will exhibit a rise in the horizontal frequency, and when l is changed the vertical frequency is likewise changed. A plot showing the two dimensional basis functions of a DCT i made on Figure 2.3.

In practical, the 2D-DCT is performed on images, by firstly splitting the image into a grid of non-overlapping blocks, typically of size 8 by 8 pixels. Then the DCT is applied for one image block at the time, until all blocks have been through the transform.

It is notable that all basis functions are fixed for a specific size $M \times N$, and therefore can be applied to any image of size $M \times N$. This is typically exploited in implementation issues, where the cosine terms can be pre-computed and stored in tables.

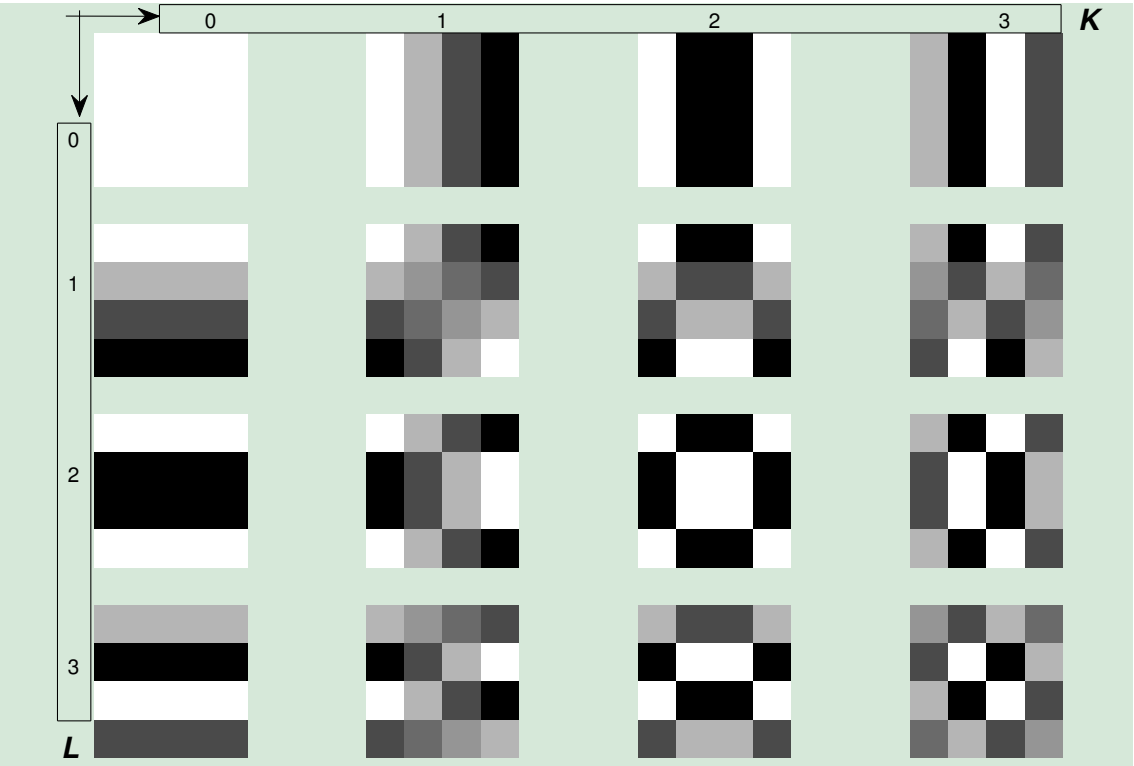


Figure 2.3. 2D-DCT basis functions with a length of $M{=}N{=}4$, shown for each instance (k,l).

2.3 Obtaining the 2D-DCT Matrix

Looking at Equation (2.6), it suggests that the DCT coefficients are obtained by running two parallel summations. However it can be written more conveniently, as a matrix-vector multiplication, by exploiting the 1-dimensional DCT basis functions, derived from Equation (2.1). The basis functions can be presented in matrix form, by evaluating all the n-values throughout the columns and the k-values down the rows, as derived in Equation (2.10).

$$D_1 = \begin{bmatrix} \cos\left(\frac{\pi(2\cdot 0+1)\cdot 0}{2N}\right)\alpha(0) & \cdots & \cos\left(\frac{\pi(2(N-1)+1)\cdot 0}{2N}\right)\alpha(0) \\ \vdots & \ddots & \vdots \\ \cos\left(\frac{\pi(2\cdot 0+1)\cdot(N-1)}{2N}\right)\alpha(N-1) & \cdots & \cos\left(\frac{\pi(2(N-1)+1)\cdot(N-1)}{2N}\right)\alpha(N-1) \end{bmatrix} \tag{2.10}$$

The 1D-DCT from Equation (2.1) can now be expressed in matrix form using the D_1 matrix derived in Equation (2.10), as shown in Equation (2.11).

$$C = D_1\vec{f} \tag{2.11}$$

where

$\vec{f}$ is any vector of length N.

In this context the input to the DCT is an image V, represented by a 2-dimensional array, hence the 2D-DCT must be applied. The 2D-DCT can also be expressed on matrix form

by utilizing D_1, as shown in Equation (2.12).

$$C = D_1 V D_1^T \tag{2.12}$$

If V is stacked column vise into the vector v, then the 2D-DCT can be represented as in Equation (2.13)

$$\text{vec}(\text{DCT2}(V)) = \text{vec}(D_1 V D_1^T) = D_2 \text{vec}(V) = D_2 v \tag{2.13}$$

Where D_2 can be described as the Kronecker product of D_1 and D_1, as shown in Equation (2.14).

$$D_2 = D_1 \otimes D_1 \in \mathbb{R}^{N \times N} \tag{2.14}$$

Multiplying D_2 by any image written on vector-form will return the DCT coefficients of the respective image.

3 Peak Signal-to-Noise Ratio

During the project a method to measure the performance of the packet loss concealment, is needed. For that task the Peak Signal-to-Noise Ratio was chosen, due to it being most commonly used method when measuring the quality of reconstruction of images. For future reference the symbolic notations of the images will be as listed below, and these will be used throughout the calculations.

- I is the original image.
- I_c is the image after the packet loss concealment.

3.1 PSNR

The PSNR is the ratio between the maximum signal value squared and the noise in the signal. In this context the maximum value is the pixel intensity of 255, as the images are represented with a bit-depth of 8 bits, and the noise is the error caused by the packet loss concealment, and is calculated by taking the Mean Squared Error(MSE) between the images I and Ic. The MSE is defined as in Equation (3.1). [5]

$$\text{MSE} = \frac{1}{m \cdot n} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (I(m,n) - I_c(m,n))^2 \tag{3.1}$$

where

$I(m,n)$ is the pixel intensity of the (m,n) pixel in I
$I_c(m,n)$ is the pixel intensity of the (m,n) pixel in I_c
M is the total number of pixels in a row
N is the total number of pixels in a column

The PSNR is usually defined in dB, as some signals have a large maximum value, MaxValue, which results in large PSNR values. The PSNR can then be described as in Equation (3.2).

$$\text{PSNR} = 10 \cdot \log \left(\frac{\text{MaxValue}^2}{\text{MSE}} \right) \tag{3.2}$$

Where MaxValue is the largest obtainable value within the signal.

4 Sparse DCT Approximation

4.1 Introduction

This worksheet will investigate the sparse discrete cosine transform approximation (SDA) as a method for performing packet loss concealment (PLC). The method relies on the discrete cosine transform (DCT), formerly described in worksheet 2, by approximating a sparse solution of the DCT coefficients. The problem is cast and solved as an optimization problem.

4.2 Methods

The basic idea behind the SDA, is to conceal visual artifacts caused by a 8×8 lost block, by exploiting the neighboring pixels of the lost block. By simplest means, all adjacent pixels to the lost block are utilized in the reconstruction. The known pixels surrounding the lost block are referred to as the frame, and has matrix representation denoted by W. Furthermore the union between the lost block and the frame is defined as V, and is illustrated in Figure 4.1.

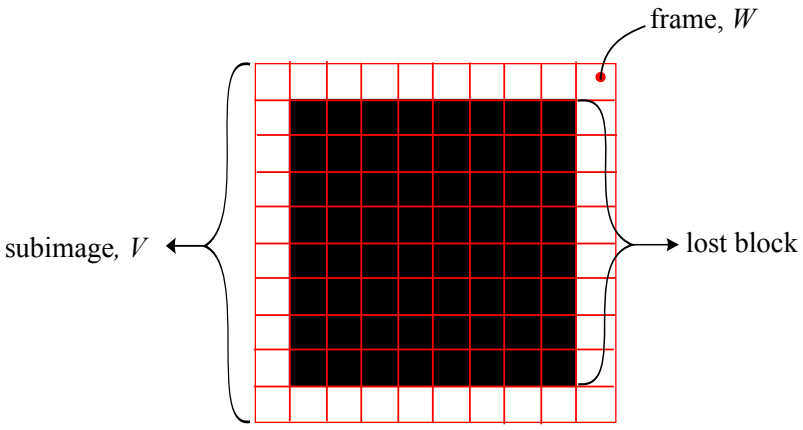


Figure 4.1. The subimage V with the known pixels W (white), and the lost block (black).

V is defined as a matrix, composed of the column vectors v_i for $i = 1\ ,\ 2\ ,\ \ldots\ ,\ n$.

$$V = \begin{bmatrix} | & | & & | \\ v_1 & v_2 & \ldots & v_n \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times n} \tag{4.1}$$

Throughout this worksheet, the sub-image will be written on vector-form:

$$v = \text{vec}(V) = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \in \mathbb{R}^{N \times 1} \tag{4.2}$$

where

$$N = n \cdot n$$

Using a fixed frame-size of one, is the standard procedure for the SDA, but it can be expanded freely according to the circumstances. Choosing a frame-size is a trade-off between several parameters such as computational complexity, the amount of information used in the reconstruction and to avoid the lost block to become insignificant compared to the frame. However in most of cases, a frame-size of one will produce a decent result of the SDA.

In particular, it was formerly shown that natural images have an approximately sparse DCT representation. However when corrupted by a lost block, the sparsity of the DCT representation will be significantly reduced, in a sub-image V. Due to the sharp edges introduced by the lost block, high frequency content is added, hence reducing the sparsity of the DCT coefficients. In order to verify the statement, a comparison is made in in Figure 4.2, where the original and a corrupted version of a sub-image are shown, with their respective DCT representation.

In order to conceal the error, it is desired to approximate a sparse solution of the DCT coefficients. During the concealment the pixels in the frame must stay unchanged, and the unknown pixels should be selected such that the DCT representation becomes as sparse as possible. The problem is cast as a minimization problem, and solved by minimizing the ℓ_1-norm of the DCT of V. Nevertheless a constraint is added to ensure that the frame is preserved in its original form. The minimization problem can then be stated as in Equation (4.3).

$$\begin{aligned} \underset{v \in \mathbb{R}^{N \times 1}}{\text{minimize}} \quad & \|D_2 v\|_1 \\ \text{subject to} \quad & Mv = w \end{aligned} \tag{4.3}$$

A binary mask matrix M, is introduced with the purpose of masking out the frame from the image vector v. In the case of a 8×8 lost block and a frame-size of one, the masking

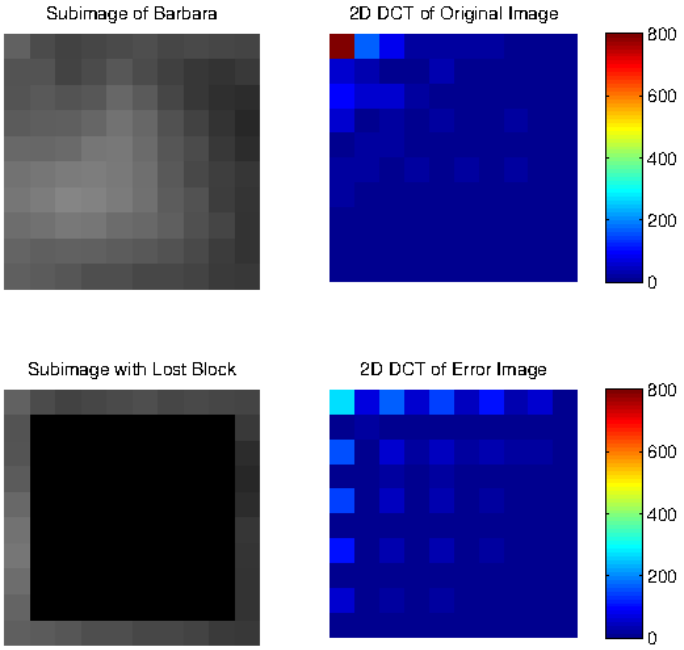


Figure 4.2. Subimages with corresponding DCT domain representations.

matrix can be defined as the block diagonal matrix shown in Equation (4.4).

$$M = \begin{bmatrix} M_1 & 0 & 0 & \cdots & 0 \\ 0 & M_2 & 0 & \cdots & 0 \\ 0 & 0 & M_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & M_{10} \end{bmatrix} \in \mathbb{R}^{100\times 100} \tag{4.4}$$

where

$$M_1 = M_{10} = I_{10} \in \mathbb{R}^{10\times 10} \tag{4.5}$$

and

$$M_i = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{10\times 10}, \quad i = 2,\ 3,\ \ldots\ ,\ 9 \tag{4.6}$$

and

I used in Equation (4.5) is the identity matrix.

Even though the definition of the masking matrix is given for a frame size of one, it is easily expanded for larger values, by inserting the same amount of identity matrices as the frame-size. Thus a frame-size of two, needs to have two identity matrices at the start and the end in the diagonal of M. For a frame-size of three, three identity matrices are used, and so forth.

By introducing the equality constraint $Mv = w$, with w being the frame written on vector-form, the frame will remain unchanged for any solution $v^\star$.

4.3 Results

The SDA method which is described in Section 4.2, has been implemented in MATLAB and it has been used to conceal two lost blocks. One block containing low frequency content and one containing high frequency content. In Figures 4.3 and 4.4 the original subimages are shown together with a corrupted version where a lost block is introduced, and one where the lost block is reconstructed using the SDA method. Together with these figures, their relative DCT representations have been plotted as well.

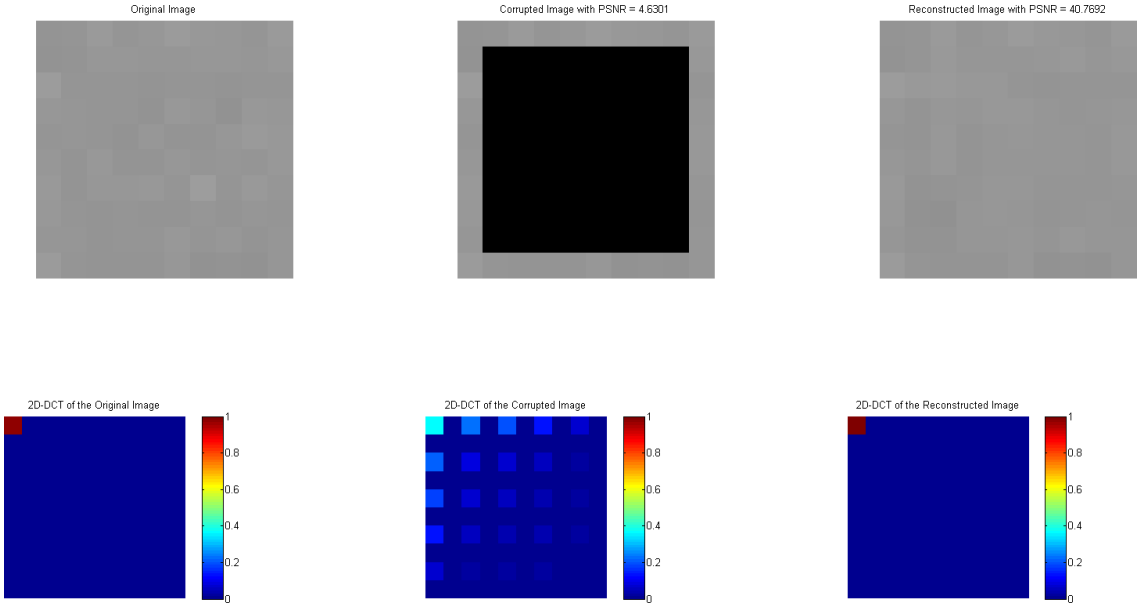


Figure 4.3. Low frequency subimages with corresponding DCT domain representations.

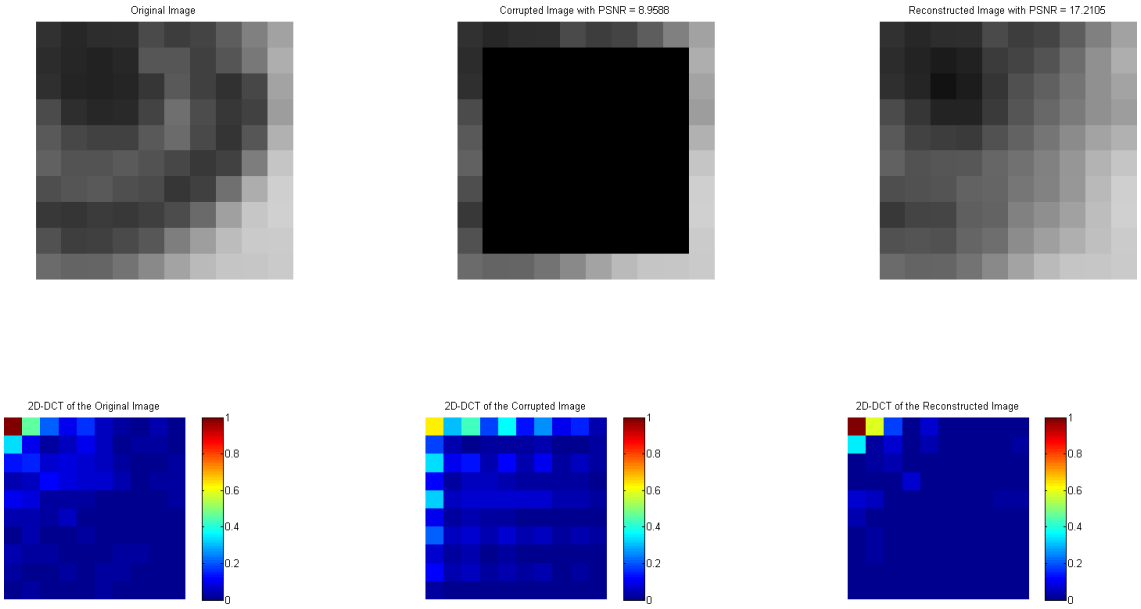
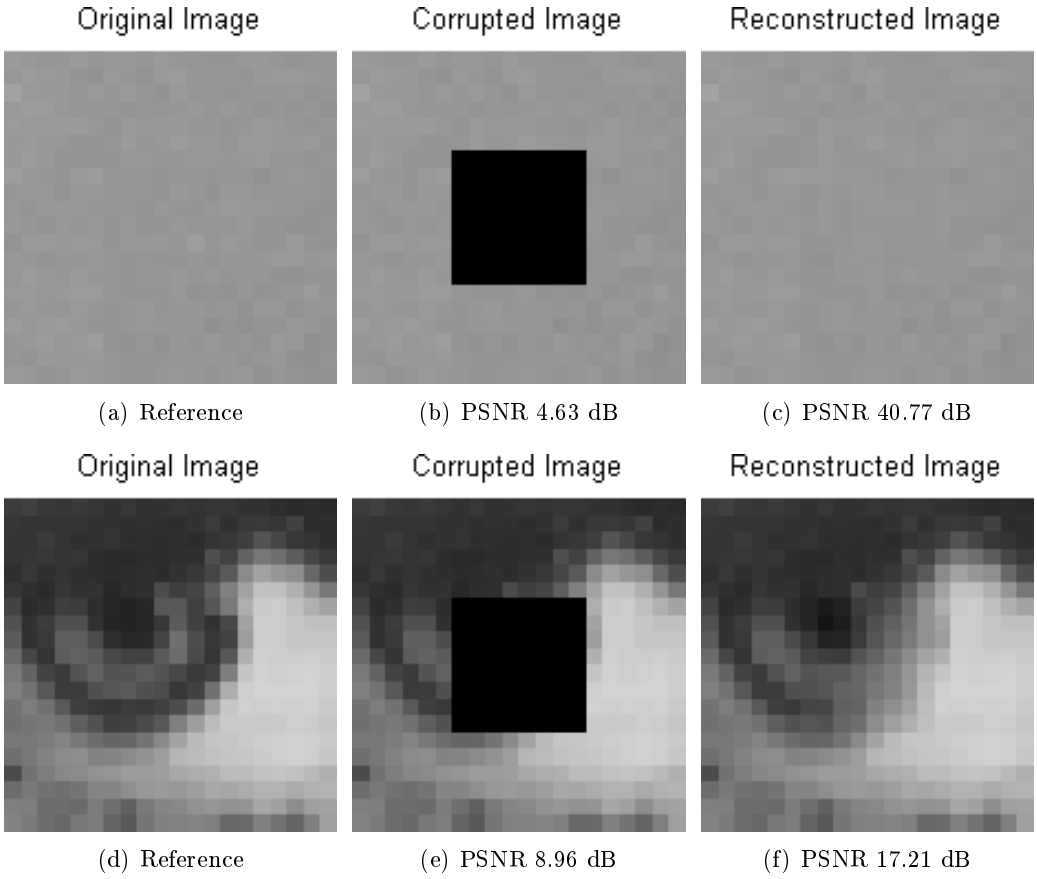


Figure 4.4. High frequency subimages with corresponding DCT domain representations.

As seen in the figures, some high frequency components are introduced in the DCT

representation, when a lost block occurs, and this was to be expected. When the concealment is done these components are minimized, which results in a reconstruction of the lost block.



(a) Reference (b) PSNR 4.63 dB (c) PSNR 40.77 dB

(d) Reference (e) PSNR 8.96 dB (f) PSNR 17.21 dB

Figure 4.5.

To make it easier to visually evaluate the concealments, zoomed plots can be seen in Figures 4.5(a) - 4.5(f). For a more objective comparison the PSNR is calculated for each 8×8 block. The low frequency subimage with a lost block, Figure 4.5(b), is improved by 36.14 dB by using the SDA method. Whereas the high frequency image, Figure 4.5(e), is improved by 8.25 dB.

5 Sparse DCT Approximation with Reweighted ℓ_1-norm

5.1 Introduction

A possible enhancement of the traditional sparse DCT approximation (SDA-1), will be investigated throughout this worksheet. The method suggests to increase the sparsity of the ℓ_1-norm, by introducing a weighting matrix to the formerly defined optimization problem in Equation (4.3), in order to ensure a more equitable penalty paid by the DCT coefficients.

5.2 Methods

As indicated in Equation (4.3), the SDA method is defined and solved as a minimization problem. Obviously the solution $v^\star$, is highly dependent on the magnitude of the DCT coefficients, as a consequence of relaxing the ℓ_0-norm in favor of the ℓ_1-norm. When minimizing the ℓ_0-norm all non-zero coefficients are treated equally, since it is a direct sparsity indicator. Whereas the ℓ_1-norm penalize large coefficients more severely than small coefficients, due to being magnitude-dependent.

To rectify the imbalance between the two norms, a suggestion found in [6] proposes a reweighting of the DCT coefficients in the ℓ_1-norm. Since the coefficients containing a significant amount of energy, are typically located in the upper left corner of the 2D DCT matrix, a method must be designed to counteract the penalty forced upon these coefficients. For this purpose a weighting matrix, Q, is introduced in the minimization problem of the SDA to apply the reweighting, and this results in a new minimization problem (SDA-RW) as seen in Equation (5.1).

$$\begin{aligned} \underset{v \,\in\, \mathbb{R}^{N\times 1}}{\text{minimize}} \quad & \|QD_2 v\|_1 \\ \text{subject to} \quad & Mv = w \end{aligned} \tag{5.1}$$

where

$$Q = \text{diag}(q_1,\ q_2,\ \dots\ ,\ q_m,\ q_{m+1},\ \dots\ ,\ q_N) \tag{5.2}$$

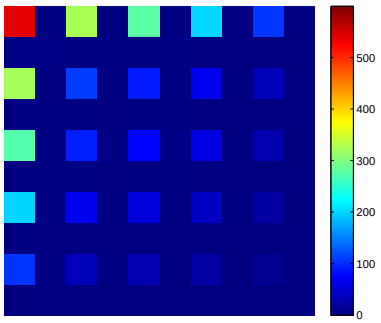
The diagonal matrix Q, has its weighting coefficients placed on the diagonal, with the first m values used for weighting the DCT coefficients to be emphasized, and the remaining are set to 1.

$$q_i = \begin{cases} < 1 & \text{for} \quad i = 1,\ 2,\ \ldots\ ,\ m \\ 1 & \text{for} \quad i = m+1,\ \ldots\ ,\ n \end{cases} \tag{5.3}$$

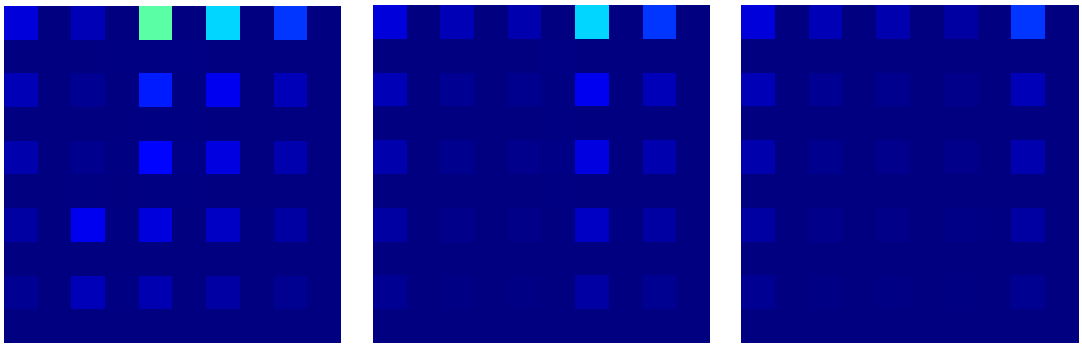
In constructing the Q matrix there are two parameters to be adjusted, they are as follows:

- Selecting the value of m
- Choosing a proper weighting coefficient, $q_1 = q_2 = \ldots = q_m$

As for the first item, it defines the percentage of DCT coefficients which are affected by the reweighting. This is typically set to a small number, as most of the energy are compacted at low frequencies. An example of this can be seen in Figure 5.1, where a 10×10 subimage containing an error is plotted for different values of m.



(a) m corresponding to 0 %

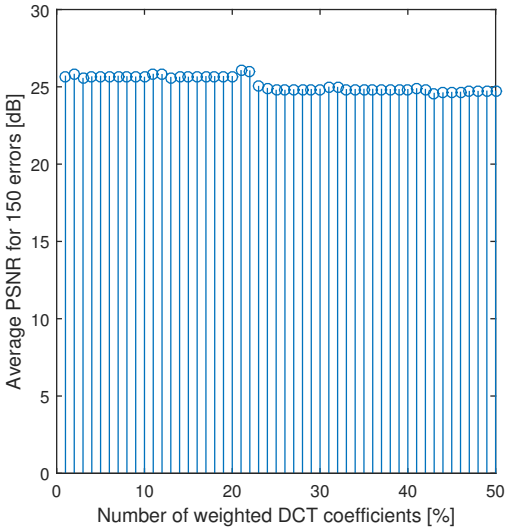


(b) m corresponding to 25 % (c) m corresponding to 50 % (d) m corresponding to 75 %

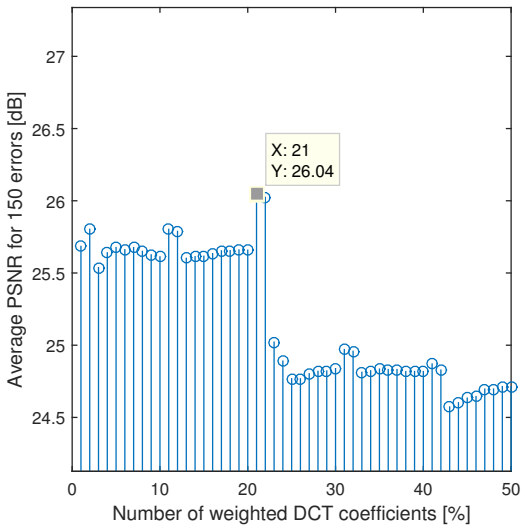
Figure 5.1. DCT domain representations of a subimage with a lost block, with values of m corresponding to a reweighting of 0 %, 25 %, 50 % and 75 % of the DCT coefficients.

Seen from Figure 5.1, a too large m value will result in reweighting coefficients which have an already low magnitude. The high frequency coefficients will eventually become insignificantly small, which draws the problem back to the starting point, where an unfair penalty is paid for low frequency coefficients. Instead the SDA-RW aims at obtaining a more evenly distributed magnitude-spectrum, as seen in the DCT representations with m equal to 25 % and 50 % of the Q matrix.

In order to have a more precise result, the PSNR value is measured against the percentage of weighted DCT coefficients in an interval of 1 % to 50 %, using a fixed value for the weighting coefficient, $q_1 = q_2 = \cdots = q_m$. The results presented in Figure 5.2, are found by averaging the PSNR over 150 reconstructed blocks.



(a) Average PSNR performance shown as a function of m with the corresponding percentage.



(b) A zoomed version of the graph shown in (a).

Figure 5.2.

Evident from Figure 5.2(b) is that 21 % exhibits the best PSNR value on average, only slightly better than that of 22 %. Therefore either of them can be subject to a subsequent implementation of the SDA-RW.

Choosing a proper coefficient value, q, is a trade-off between choosing a too large value which does not contribute to a more equitable penalty distribution, and a too small values that almost eliminates the coefficients.

Through simulations, it was observed that the weighting coefficients which exhibited the best performance of the PSNR, frequently occurred in the interval from 0.25 to 0.50. However the performance of the distinct values in the interval are somewhat equal, hence choosing the midrange of the interval, is an obvious choice.

5.3 Results

In conclusion of the analysis conducted in the former section, a weighting matrix has been characterized with the following parameters: an m value which reweights 21 % of the DCT coefficients, and the weighting coefficient q_1 , ... , $q_m = 0.37$.

To get a brief insight into the performance of the SDA-RW, a concealment is performed in both the eye of *Lena* and in the background of the *Lena* image. The results are presented in Figure 5.3, with a comparison to the SDA-1 method.

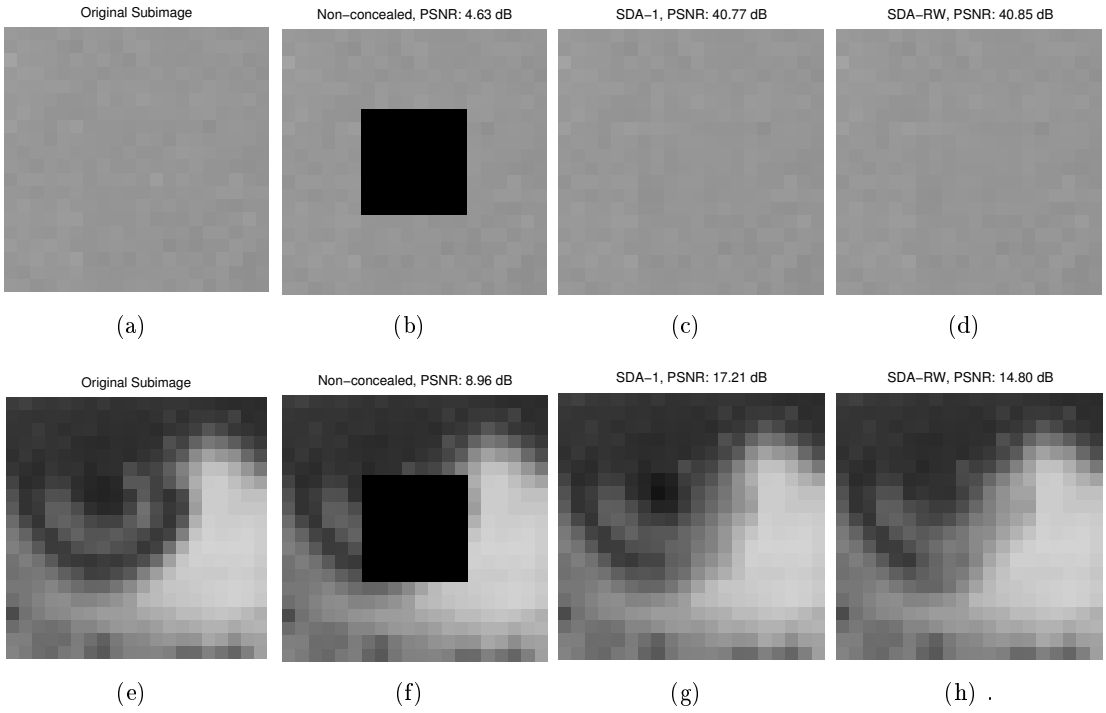


Figure 5.3. Reconstruction of a lost block in a low- and high-frequency area of the *Lena* image, using the SDA-1 and SDA-RW methods

The first error block presented in Figure 5.3(b), is located in an area with only small variations. From the results shown in Figure 5.3 (c) to (d), no observable difference is noticed, neither by visual inspection or in terms of the PSNR. However both methods manages to improve the PSNR of the lost block with more than 36 dB. As for the other image in Figure 5.3(e)-(h), a worsening of the quality in the reconstruction occurs, as the DCT coefficients in the ℓ_1-norm are weighted.

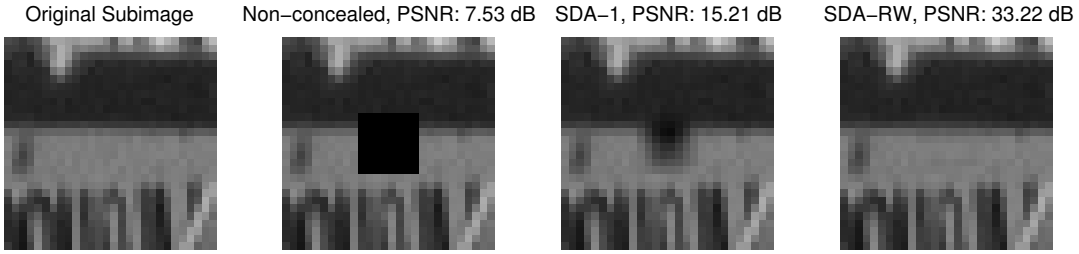
In order to verify whether the SDA-RW offers any improvement compared to the traditional SDA method, a more thorough comparison is carried out, where the performance of the two methods are evaluated on the four images: *Lena*, *Barbara*, *Boat* and *Cameraman*, with 150 error blocks introduced to each image. For each error block, the PSNR for the reconstructed block is calculated. Finally the PSNR values are averaged over all of the error blocks. The results from the comparison are shown in Table 5.1.

PSNR	SDA	SDA-RW	Improvement
Average	28.88	30.32	1.44
Lena	30.35	31.28	0.93
Barbara	26.00	27.48	1.48
Boat	25.39	26.59	1.20
Cameraman	33.76	35.91	2.15

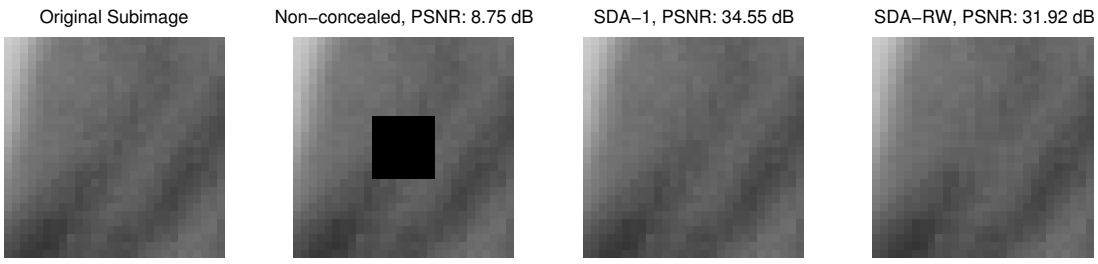
Table 5.1. SDA-RW measured against the traditional SDA method in terms of the PSNR listed in dB.

5.4 Discussion

As seen from the Table 5.1, an improvement of 1.44 dB in the PSNR can be expected, when using SDA with reweighting (SDA-RW) rather than the SDA-1. Even for the *Cameraman* image, the gain in PSNR is above 2 dB on average. Whereas for some images the performance of the SDA-RW is not quite as good. Taking a closer look into the PSNR value of the distinct reconstructed blocks, reveals no clear tendency of the improvement by using reweighting, on the contrary a lot of fluctuations occur. In some of the most extreme cases, the PSNR value benefits greatly from the reweighing, with improvements up to 18 dB compared to the SDA-1. However it also occurs that the reconstruction is worsened from the reweighting, thereby degrading the PSNR value. Two of such cases are presented in Figure 5.4(a) and 5.4(b), with their respective PSNR values presented.



(a) Extreme case, with a good SDA-RW performance.



(b) Extreme case, with a bad SDA-RW performance .

Figure 5.4.

Seen from the case shown in Figure 5.1, the SDA-RW handles edges in the lost block significantly better than the SDA-1, this is equally valid in other inspected error-blocks which encounters an edge. Furthermore all visual artifacts caused by the SDA-1 concealment, are efficiently obscured after reweighting. In other cases where the lost block intersects several edges, will however significantly worsen the reconstruction in terms of PSNR, an example of this was shown in the initial test in Figure 5.3(g) and 5.3(h). This is generally the case when dealing with high frequency blocks.

The overall results from the comparison between the SDA-1 and the SDA-WS, indicates that improvements can be obtained, by introducing a weighting matrix. Potentially further improvements can be obtained, by launching a more exhaustive investigation into the parameters of the Q matrix. That could be simulating the average performance for all

possible values of the weighting coefficient at all possible m-values, and picking the best one. However, the SDA-RW method will still be subject to the weakness in handling high frequency content.

All the proposed methods and results are based upon the SDA using a frame-size of 1 pixel-width. Considering a 8×8 lost block containing 64 unknown pixel values, and a frame of 36 pixels known pixels, there seems to be an unbalance between the two. Thus a suggestion is to include more known information into the concealment problem, by expanding the frame such that it covers a larger neighborhood area, and thereby possibly obtaining a better reconstruction. Choosing an optimal frame-size is not a trivial task, therefore it will be subject to further investigation, where the relation between quality and frame-size is analyzed.

5.5 Conclusion

In this worksheet a method for deriving a reweighting of the ℓ_1-norm of the DCT coefficients was suggested. The average performance of the SDA-RW exhibited an average quality improvement of 1.44 dB in the PSNR, compared to the SDA-1 method. In order to clarify the reason behind the fluctuations in the PSNR improvement or degrading by using reweighting, two "extreme scenarios" was considered. It was clear both by visual inspection and by the PSNR quality measure that the SDA-RW has the potential to enhance the performance of the SDA-1, especially in cases where the lost block intersects an single edge. However, the SDA-RW method can also cause a degradation of the performance of the traditional SDA. This is often valid, when the error block encounters multiple edges. For further investigation, it was proposed to exhaustively examine all parameters of the weighting matrix as well as analyzing potential improvements by expanding the frame of the lost block.

6 Frame-size Selection of the SDA Through an Experimental Approach

6.1 Introduction

This worksheet describes the investigation of finding the best average performing frame-size for the Sparse DCT Approximation(SDA), this is done with an experimental approach. The SDA was defined to have a frame-size of 1 pixels width by default, but no investigation was made to see if other sizes would produce better results. This is an attempt to improve the average performance of the SDA, by looking into other frame-sizes.

6.2 Methods

The SDA method was designed with a static frame-size of 1 pixels width, however in this worksheet experiments will be made to test the performance of the SDA with other frame-sizes. The error-concealment is done based on the data contained in the frame, so by expanding it, more data becomes available. This increase in available data should in theory result in a better concealment. However if the frame is expanded too much, then it might end up containing data which is no longer relevant to the reconstruction of the lost block, and this will end up worsening the reconstruction.

The performance of different frame-sizes will be evaluated, but firstly the desired frame-size interval should be determined. It was chosen based on the evaluation of different characteristics of the SDA. The first one being the case explained earlier where the frame gets too big, and at some point ends up including undesired data. The second one was the computational complexity of the SDA as the frame-size is increased. Both cases makes larger frame-sizes undesirable, so for this analysis a maximum frame-size of 12 pixels width was chosen. So the SDA's performance will be analyzed with the frame-sizes from 1 to 12.

In this analysis the 4 standard test images will be used, and for each image 150 random packet losses will be introduced, and for the purpose of the investigation the location of these is assumed to be known. Each packet loss will then by reconstructed with the different frame-sizes, and then the PSNR of the reconstructed block will be

calculated. Then the average PSNR for each frame-size will be computed together with a histogram of where the amount of times a given frame-size was the optimal one.

6.3 Results

A histogram of the frame-sizes with optimal PSNR was made, in order to see which frame-size got the highest occurrence of optimal PSNR. This histogram can be seen in Figure 6.1, and a table of the exact values can be seen in Table 6.1. Where in the table the highest occurrence for each image is marked with green, and the lowest with red.

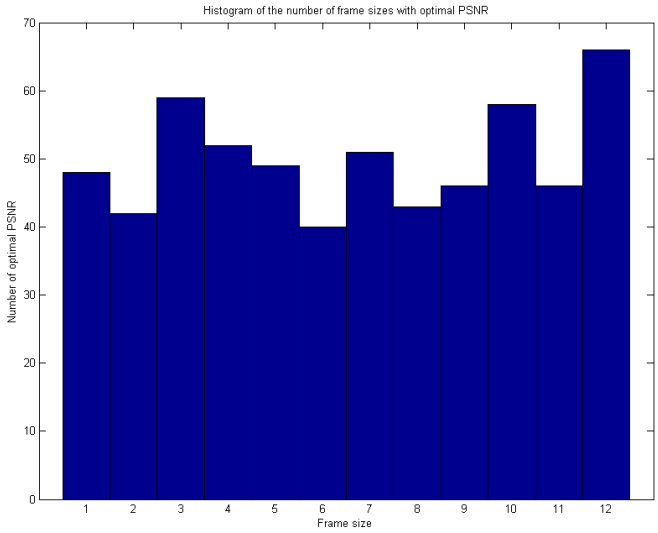


Figure 6.1. Histogram of the frame-sizes with optimal PSNR

Frame size	1	2	3	4	5	6	7	8	9	10	11	12
Lena	15	8	16	15	13	8	13	11	18	15	9	9
Barbara	8	12	7	8	10	7	13	12	11	21	13	28
Boat	15	13	17	18	11	10	13	13	8	10	6	16
Cameraman	10	9	19	11	15	15	12	7	9	12	18	13
Total	48	42	59	52	49	40	51	43	46	58	46	66

Table 6.1. Histogram values of the frame-sizes with optimal PSNR

As seen in the histogram all frame-sizes are almost equally represented, but with a size of 12 having the most occurrences. These result indicates that selecting a static frame-size might not be the best solution, so some investigation into creating a dynamic frame should be done.

Besides the histogram the average PSNR for each frame-size was also computed, in order to assist in the selection of a static frame-size. These PSNR values can be seen in Figure 6.2, and in Figure 6.3 the improvements with respect to SDA with a frame-size of 1 can

be seen. A table containing all the values for each image can be seen in Tables 6.2 and 6.3. In this table the best averages are displayed with green and the worst with red.

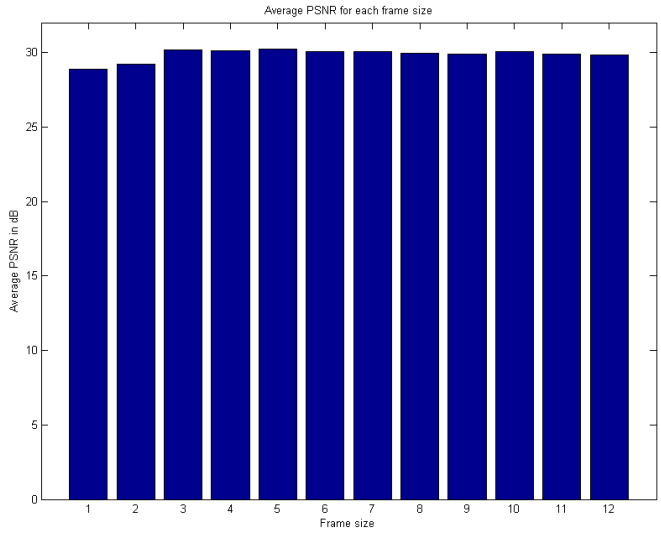


Figure 6.2. Average PSNR in dB for each frame-size.

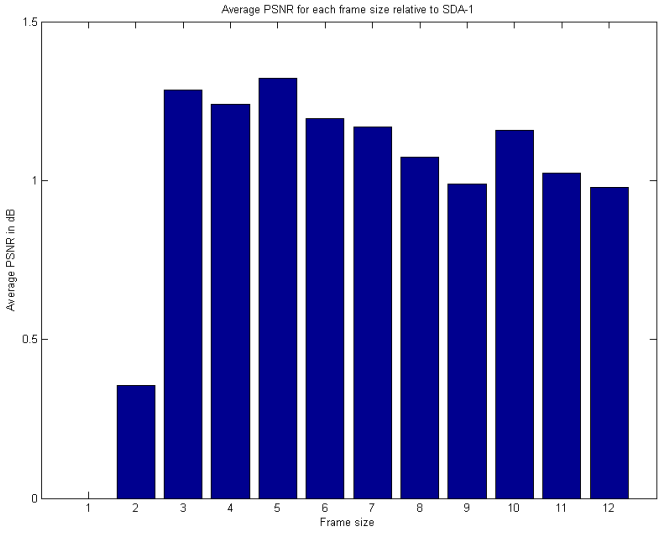


Figure 6.3. Average PSNR improvement with respect to SDA_1 in dB for each frame-size.

Based on the average PSNR alone the static frame-size should be set to 5, and not 12 like the case for the histogram results, from Figure 6.1.

6.4 Discussion

Since no direct conclusion can be derived from the two figures, the histogram (see Figure 6.1) and the average PSNR (see Figure 6.3), the selection of frame-size must be a

Frame size	1	2	3	4	5	6
Average	28.88	29.23	30.16	30.12	30.20	30.07
Lena	30.35	30.38	31.08	30.94	30.96	30.98
Barbara	26.00	26.91	27.94	28.08	28.51	28.34
Boat	25.39	25.38	26.38	26.36	26.23	26.15
Cameraman	33.76	34.25	35.25	35.09	35.10	34.82

Table 6.2. Average PSNR's in dB for each frame-size

Frame size	7	8	9	10	11	12
Average	30.04	29.95	29.87	30.03	29.90	29.86
Lena	31.05	30.87	30.94	30.89	30.81	30.85
Barbara	28.38	28.46	28.48	28.79	28.44	28.57
Boat	26.16	26.12	25.93	26.21	25.92	26.03
Cameraman	34.58	34.35	34.12	34.25	34.44	33.97

Table 6.3. Average PSNR's in dB for each frame-size

compromise between both results. The histogram resulted in a size of 12 being the best solution, where the average PSNR measurements resulted in 5.

By looking at both the histogram and the average PSNR it can be seen than selecting a frame-size of 12 will result in a worse average PSNR than for other frame-sizes. So both the average performance and the computational complexity can be improved by selecting a smaller frame-size. Then by looking at the second most represented frame-size in the histogram, which is 3, it can be seen that the average PSNR is good, only being 0.06 dB lower compared to the best average PSNR for the frame-size of 5. So based on these results the best solution for a static frame-size would be 3, as it performs well both in average PSNR measurements and it has a high occurrence in the histogram, meaning the size of 3 is the optimal frame-size more often than other frame-sizes.

6.4.1 Oracle Dynamic Frame Selection

The histogram from Figure 6.1 showed an almost equal representation of each frame-size, which means that not one frame-size had a dominant appearance. Selecting a static frame-size will result in the SDA not performing optimally in most cases. So in this section a quick investigation will be done, to see how much of an improvement in the average PSNR can be obtained. This is done by using an Oracle function, which reconstructs each packet loss with the optimal frame-size, instead of a fixed frame-size for all packages. The average PSNR of the reconstructed blocks with a static frame-size, and the average PSNR of the dynamic oracle frame can be seen in Table 6.4.

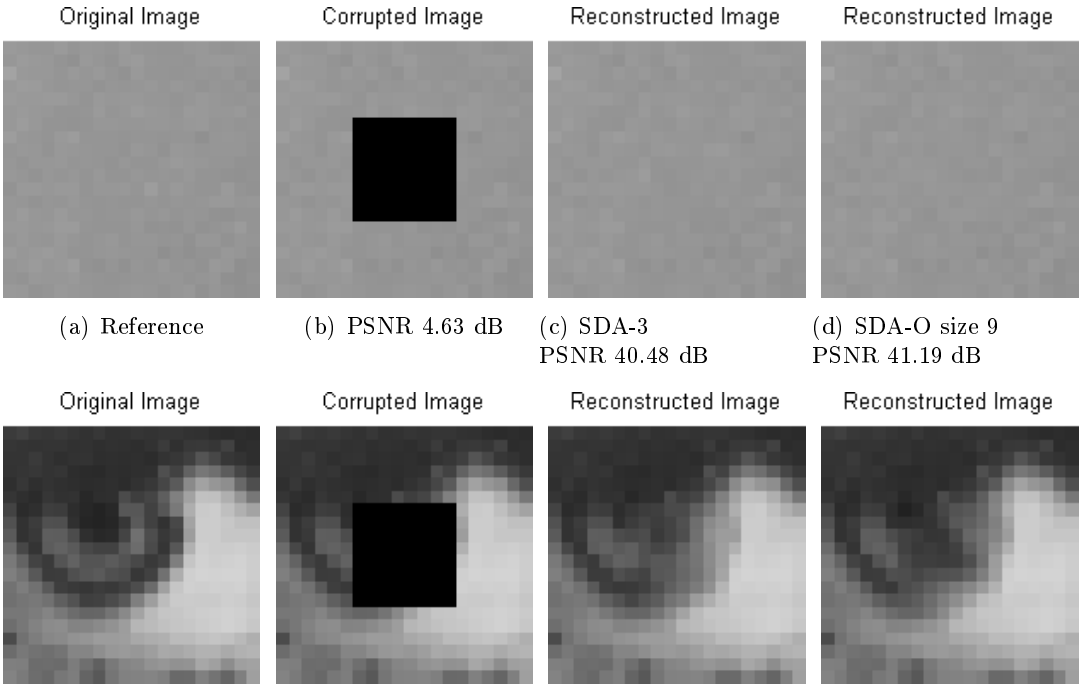
	SDA-3	SDA-O	Improvement
Total	30,24	33	2,76
Lena	31,66	34,16	2,5
Barbara	27,94	31,36	3,42
Boat	26,38	28,59	2,21
Cameraman	35,25	38,27	3,02

Table 6.4. Obtainable improvements in PSNR in dB by using a dynamic frame-size

The table shows an average improvement of 2,76 dB by using the dynamic frame compared to a static frame. So looking in to a way of making an algorithm which can create a dynamic frame might be a good idea, since an improvement of almost 3 dB can be obtained by using one.

6.5 Concealment Results

This section is just to show some concealments using the SDA-3 and the SDA-O, in order to make visual inspection of them. The SDA-3 concealments can be seen in Figures 6.4(c) and 6.4(g), and the SDA-O can be seen in Figures 6.4(d) and 6.4(h).



(a) Reference (b) PSNR 4.63 dB (c) SDA-3 PSNR 40.48 dB (d) SDA-O size 9 PSNR 41.19 dB

(e) Reference (f) PSNR 8.96 dB (g) SDA-3 PSNR 17.11 dB (h) SDA-O size 12 PSNR 19.17 dB

Figure 6.4. Standard Test Images

6.6 Conclusion

The purpose of this worksheet was to investigate the size of the frame used in the SDA method, using an experimental approach, and to see if the performance could be improved by selecting another frame-size than the default size of 1. Tests were made on 4 images with 150 random package losses introduced in each. These were concealed and the PSNR for each concealment was calculated. Based on these results a frame-size of 3 was selected as the static frame-size which is to be used in the SDA, for the best average performance.

It was also discovered that selecting a static frame-size was not the best solution, since the lost blocks had different optimal frame-sizes. So a small test were made in order to see how much could be gained by creating a dynamic frame, and it turned out that there was much to gain. For further improvement of the SDA, a dynamic frame algorithm could be looked in to.

7 SDA with Dynamic Frame using Patch Search

7.1 Introduction

The error concealment of lost blocks is carried out using the sparse DCT approximation (SDA) that by default uses a fixed frame size of one (SDA-1). However it has been proven that the choice of frame size is crucial for the quality of the restored image block.

The SDA-Oracle(SDA-O) method described in worksheet 6, is an upper bound indicator of the SDA performance using a dynamic frame. It has proven that gains up to 3 dB in the PSNR is obtainable, compared to the SDA using the best performing fixed frame size. The patch search algorithm seeks to optimize the SDA-1 method, by approximating the optimal frame size for each error block. The concealment method utilizes a subset of known pixels in the neighborhood of the lost block, also denoted a *patch*, in order to find a similar structure across the entire image or simply a subset of the image. The patch search procedure eventually returns the best fit for the original pixels in the lost block, which is later used to find the optimal frame size.

7.2 Methods

In order to extensively exploit the full potential of the SDA, a practical applicable method is suggested for this purpose. The SDA-patch search (SDA-PS) method aims at creating a decision rule, of what frame size to use in the PLC via the SDA. However this is not a trivial task since the original image block is not accessible. Instead the patch search algorithm finds a block which is assumed to be similar to that of the lost block. An image denoted I, has a pre-defined search region S around the lost block, containing a specific amount of known pixels. The vector $\mathbf{y}'$ also presented in Figure 7.1, denotes a region containing both a packet loss and the associated frame surrounding the lost block, where $M_{\hat{\mathrm{y}}}$ is the number of unknown pixels and M_{y} is the number of known pixels.

$$\mathbf{y}' = [\mathrm{y}'_1, \mathrm{y}'_2, \ \dots \ , \mathrm{y}'_{M_{\hat{\mathrm{y}}}+M_{\mathrm{y}}}]^{\mathsf{T}} \tag{7.1}$$

$\mathbf{y}'$ is also denoted as the union between the lost block and the pixels in a neighborhood of the lost block.

$$\begin{aligned} \mathbf{y}' &= \mathbf{y} \cup \hat{\mathbf{y}} \\ \mathbf{y} &= [\mathrm{y}_1, \ \dots \ , \ \mathrm{y}_{M_\mathrm{y}}]^\intercal \\ \hat{\mathbf{y}} &= [\hat{\mathrm{y}}_1, \ \dots \ , \ \hat{\mathrm{y}}_{M_{\hat{\mathrm{y}}}}]^\intercal \end{aligned} \tag{7.2}$$

The method works under the assumption that the known frame surrounding the lost block $\mathbf{y}$, is an indication of what is contained within the block. Hence a search for a similar patch is conducted within the area of S, which makes use of the mean square error (MSE) to find the best fit. The best fitting patch $\mathbf{z}'$ is defined as in Equation (7.3), and an illustration of this can be seen in Figure 7.1.

$$\mathbf{z}' = \mathbf{z} \cup \hat{\mathbf{z}} \tag{7.3}$$

The MSE defined in Equation (7.4), measures the similarities between the two patches written on vector-form.

$$\varepsilon = \sum_{i=1}^{n} (z_i - y_i)^2 \quad \text{for} \quad i = 0, \ 1, \ 2, \ \dots \ , \ n-1 \tag{7.4}$$

where

y_i is the i^{th} pixel-value in the patch frame
z_i is the i^{th} pixel-value in the patch frame

A principled illustration of the search method is presented in Figure 7.1, where the patches $\mathbf{y}'$ and $\mathbf{z}'$ are depicted within a search region of the Lena image.

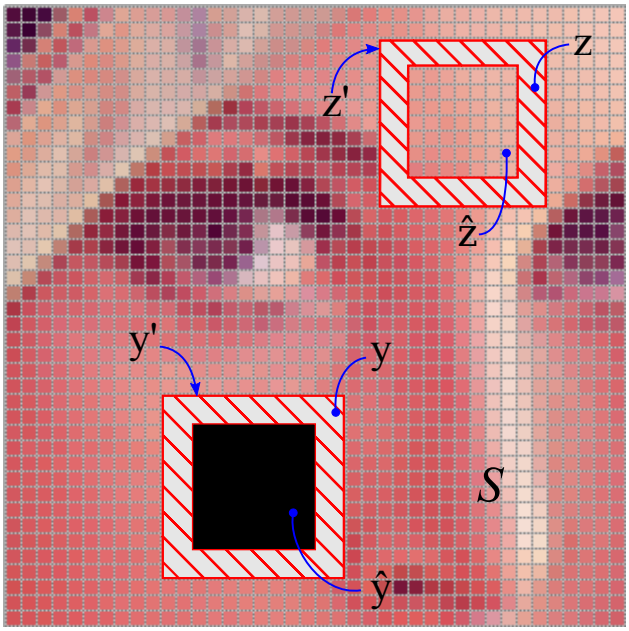


Figure 7.1. A lost block is shown for the Lena image, with the associated patch y and the best patch fit z.

The known pixels enclosed by $\mathbf{z}$, are expected to be highly correlated with the original data of the lost block. Thus the optimal frame size found for $\mathbf{z}'$, is exploited in the reconstruction of the lost block, and is easily found in terms of the PSNR. By brute-force the SDA algorithm runs through all frame sizes in a predefined interval, and selects the one which maximizes the PSNR. The found frame size, is the one expected to carry out the best packet loss concealment, and therefore directly applied in the reconstruction of $\hat{\mathbf{y}}$.

A fundamental uncertainty in using the patch search algorithm, is that it relies solely on a single patch fit. To reduce this potential uncertainty, a search for several best fitting patches can be carried out. They must be ordered in a prioritized manner, where only the N best patches $\mathbf{z}_{i=1,2,\dots,N}$ are considered, and thereby neglecting patches of lower fit. The selected patches are combined by two different methods, one utilizing the *median* and the other using an *average* approach.

7.2.1 SDA Patch Search using Median and Average Approach

The SDA patch search (SDA-PS) seeks to improve the traditional SDA method, by introducing a dynamic frame size. Using a found estimation of the missing block, it targets the one specific frame-size, which optimizes the PSNR value of the lost block. The enhancement of the SDA-PS presented in this section, suggests that multiple patches are found, in case the best fitting $\hat{\mathbf{z}}$ is not a good approximation of $\hat{\mathbf{y}}$.

In accordance with the number of best fitting patches included in the search algorithm, the same number of optimal frame-sizes will be returned. This is denoted on vector-form in Equation (7.5), in order of priority with the first element given by the best patch fit, the second element given by the second best patch fit etc., of a total amount of N patches.

$$\mathbf{f} = [\mathrm{f}_1, \mathrm{f}_2, \dots, \mathrm{f}_N] \tag{7.5}$$

where

f_i is the optimal frame size found in $\mathbf{z}_i$

Given the vector $\mathbf{f}$, it is straightforward to find the frame-size defined by using the average approach $\mathbf{f}_{avg}$ and the median approach $\mathbf{f}_{med}$.

$$\mathbf{f}_{avg} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{f}_i \tag{7.6}$$

$$\mathbf{f}_{med} = \mathrm{median}(\mathrm{f}_1, \dots, \mathrm{f}_N) \tag{7.7}$$

In the next section, all the three variations of the SDA patch search algorithm: SDA-PS, SDA-PS average (SDA-PS-A) and SDA-PS median (SDA-PS-M), are compared against the performance of the SDA-3 and SDA-O, to confirm whether it serves its purpose or not.

7.3 Results

A MATLAB implementation of the three SDA-PS methods has been conducted, to get an intuition of their performance. Concealments are done in two areas of the *Lena* image, which can be seen with their respective results in Figure 7.2 and 7.3. All the results are obtained by taking the three best fitting patches into account.

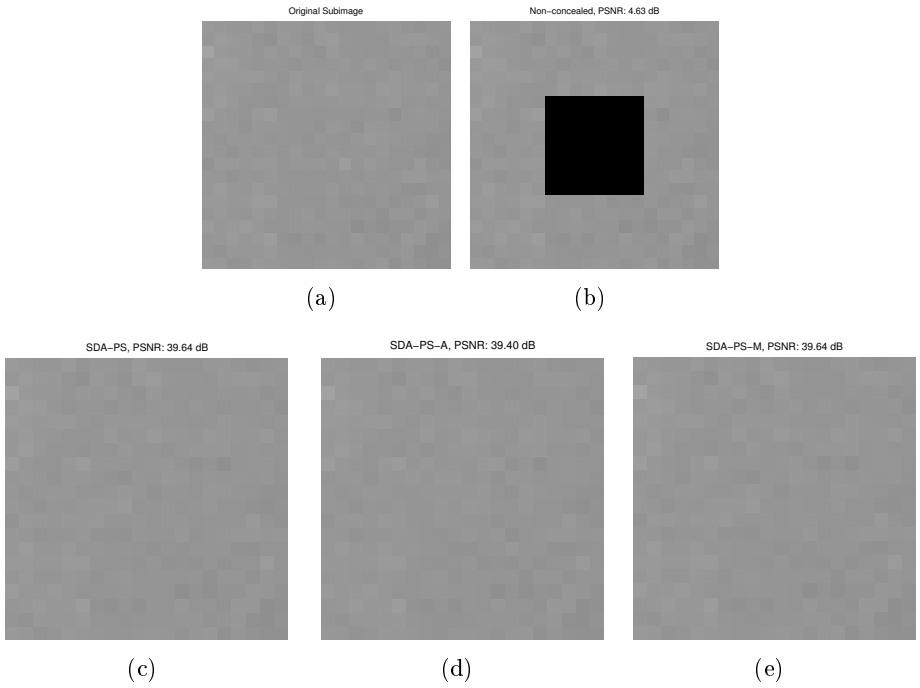


Figure 7.2. Low frequency block reconstruction.

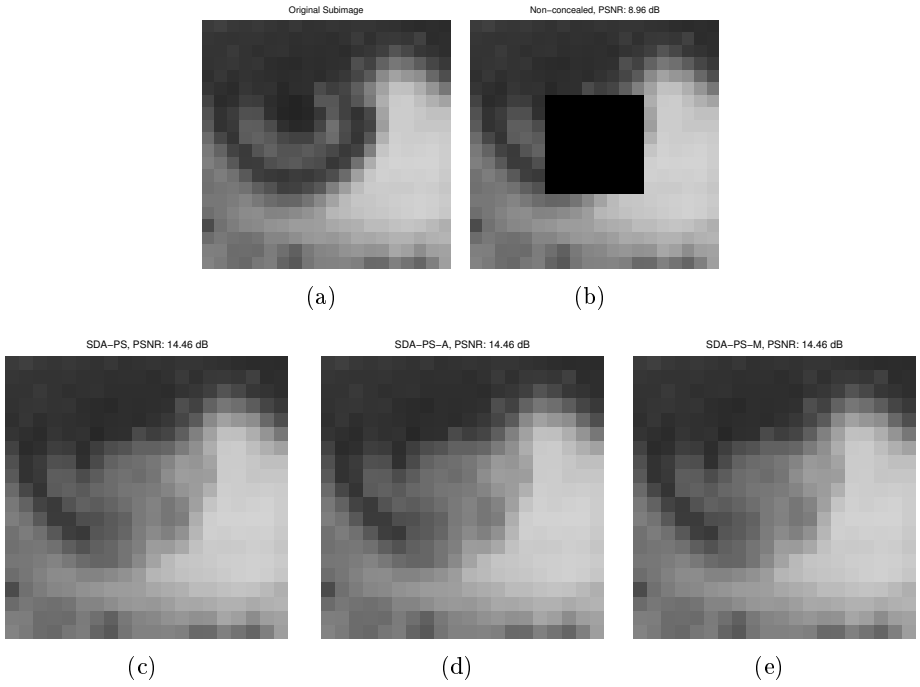


Figure 7.3. High frequency block reconstruction.

As seen from the results in Figure 7.2, all three variations of the SDA-PS carries out a decent concealment of the lost block, which leaves no visual artifacts from the concealment. The PSNR values of all the reconstructed blocks are very similar, due to the optimal frame size found in the three cases are equal, except for the *median* approach.

The second image presented in Figure 7.3(a)-(e), is placed in a area with a lot of details in it, and thereby impeding the reconstruction of the lost block. This is also reflected in the concealment results, where the outermost edge of the eyeball is hardly reconstructed, but merely smeared out in the rest of the eye.

The results from the SDA-PS and its variations will next be compared against the SDA with a fixed frame size, using a more comprehensive approach. A test is done using four standard images, with 150 errors simulated for each image. Each packet loss will be reconstructed in accordance with the above methods, and the PSNR value is calculated respectively. All results are summed up in Table 7.1.

PSNR	SDA-3	SDA-PS	SDA-PS-A	SDA-PS-M	SDA-O
Average	30.16	30.48	30.51	30.48	32.95
Lena	31.08	31.29	31.44	31.40	33.58
Barbara	27.94	28.90	28.81	28.90	31.36
Boat	26.38	26.57	26.55	26.48	28.60
Cameraman	35.25	35.17	35.23	35.14	38.27

Table 7.1. Results presented for the SDA-PS and its variations, compared against the fixed frame-size SDA and the SDA oracle (SDA-O).

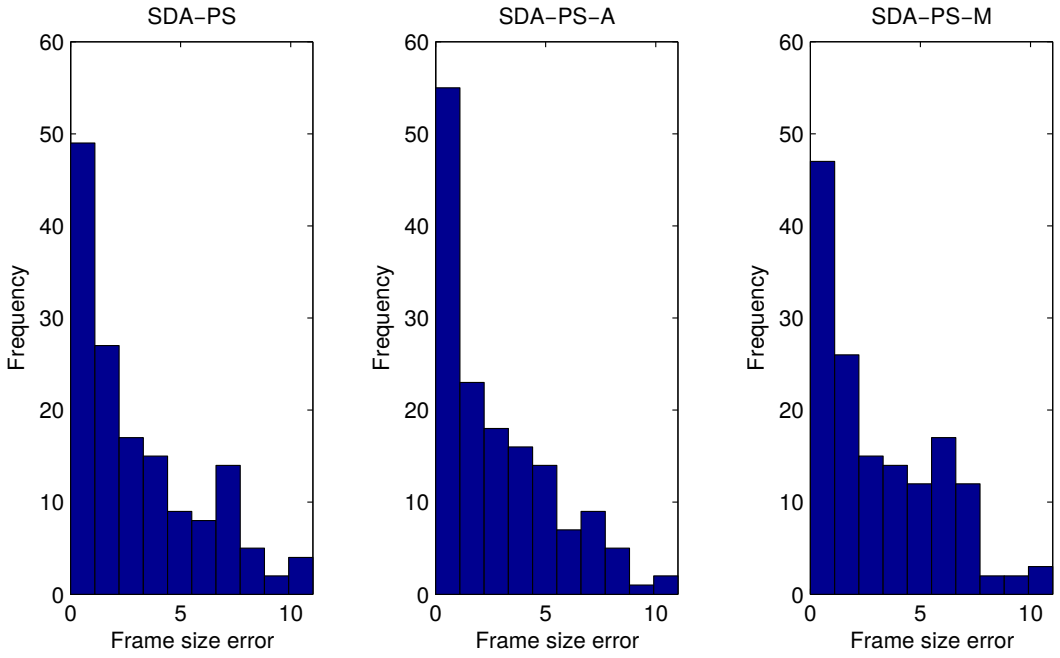
Both the results from the SDA-PS-A and the SDA-PS-M are obtained by taking the three best fitting patches into consideration.

7.4 Discussion

The results presented in Table 7.1, only shows a moderate improvement of the SDA-PS, compared to the SDA-3. Also the *average* and *median* approach of the SDA-PS, showed no significant improvement, but has a similar performance to that of the standard SDA-PS. However, since the SDA-O reveals that improvements around 3 dB in the PSNR are obtainable, the SDA-PS and its variations have not exploited the full potential of using a dynamic frame-size.

The reason behind the moderate performance of the SDA-PS, is possibly caused by the fact that it seeks to approximate the optimal frame-size of the lost block, without considering the consequences of selecting a non-optimal frame-size. In Figure 7.4, a histogram is presented based on the *Cameraman* image with 150 reconstructed blocks, it depicts the frequencies at which deviations from the optimal frame-size occurs.

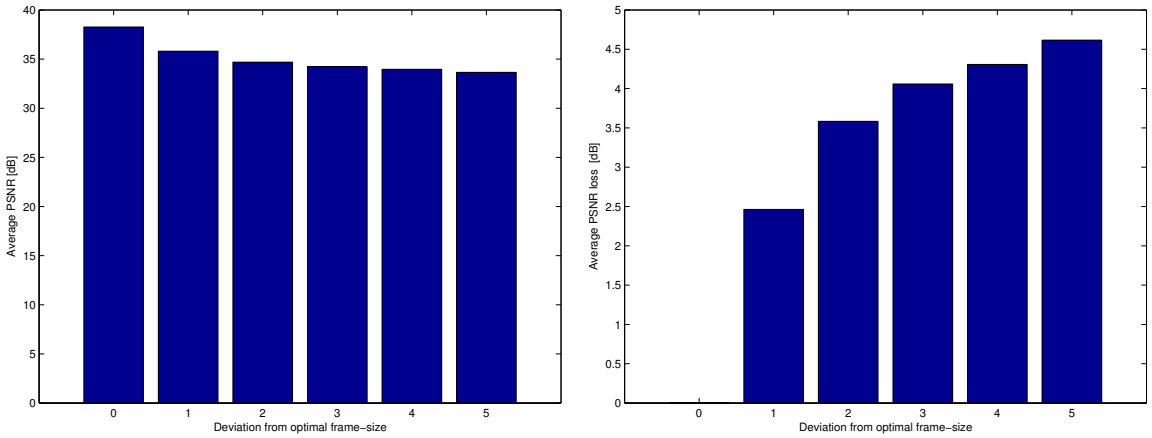
Even though the SDA-PS does not consistently hit the exact value that maximizes the PSNR, it may suggest a frame-size with only a small deviation. This is the case shown in Figure 7.4, where the majority of all estimated frame-sizes lie in a nearby interval of



(a) (b) (c)

Figure 7.4. Histograms plotting the frequency for the occurrence of distinct frame-size errors, for the three methods: SDA-PS, SDA-PS-A and SDA-M

the optimal frame-size. Unfortunately, reconstructing the lost block with a near-optimal frame-size, can have a crucial impact on the PSNR value. A brief experiment is conducted in order to verify, the degrading effect by deviating only a small amount from the optimal frame-size. Based on the *Cameraman* image the average PSNR value is plotted in Figure 7.5(a), as a function of consistently deviating from the optimal frame-size by a specific number. In the second plot presented in Figure 7.5(b), all values are calculated relative to a deviation of zero, thereby emphasizing the loss in the PSNR that the distinct frame-size errors are subject to.



(a) Average PSNR plotted as a function of deviation from the optimal frame size.

(b) Average PSNR loss is shown, relative to a deviation of zero(exact estimate).

Figure 7.5.

It is evident from Figure 7.5(b), that differing from the optimal frame-size by one in all error blocks, the average PSNR is subject to a loss of approximately 2.5 dB, and the tendency is steadily growing for larger frame errors. Hence increasing the overall performance of the SDA-PS requires a higher accuracy rate for the frame-size selection, which is not acquired by neither the SDA-PS-A or the SDA-PS-M. A potential improvement of the SDA-PS could be altering the patch search algorithm, so it no longer makes use of the MSE, but instead the structural similarity index measurement (SSIM)[6], when comparing the similarities between the patches. The drawback of the MSE is that it only takes the error between each individual pixel into consideration. Whereas the SSIM considers the luminance, contrast and structure, of the whole patch. Therefore it is expected that the SSIM is better at evaluating the similarities of the patches.

Earlier the concept of introducing reweighting into the SDA method, was proven to have a positive outcome. The use of reweighting can be generalized to concealment problems using any frame-size, by adapting the parameters of the Q matrix to the size of the frame. Based on the results of the reweighting experiment, a combination with the SDA-PS, should be investigated.

7.5 Conclusion

Throughout this worksheet the PLC method denoted as SDA patch search (SDA-PS) has been investigated and implemented. The SDA-PS seeks to create a decision rule, for the choice of frame-size, in the SDA. The performance of the method was shown to be slightly better than the fixed frame approach with the best performance , SDA-3, and even performed 1 dB better in of the tested images. However the developed algorithm did not manage to exploit the full potential of the SDA using a dynamic frame, and is therefore subject to further research. One potential improvement suggests the use of other methods, e.g. the SSIM for measuring similarities between two patches. Furthermore a combination of the reweighting and the SDA-PS was suggested, but that will further require an analysis into the behaviour and the design of the Q matrix in larger dimensions.

Bibliography

[1] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 3rd edition, p. 580, Pearson Education, 2008.

[2] J. W. Woods, Multidimensional Signal, Image, and Video Processing and Coding, 2nd edition, p. 504, Academic Press, 2011.

[3] S. A. Khayam, The Discrete Consine Transform(DCT): Theory and Application, URL:http://www.cse.iitd.ernet.in/pkalra/siv864/assignment2/DCT-TR802.pdf. Download Date: 08/11-2014.

[4] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 3.rd ed. Pearson Education, 2008.

[5] MathWorks, PSNR, URL:http://www.mathworks.se/help/vision/ref/psnr.html?refresh=true. Download Date: 01/11-2014.

[6] E. J. Candés, M. B. Wakin, and S. P. Boyd., Enhancing Sparsity by Reweighted ℓ_1 Minimization, Journal of Fourier Analysis and Applications, 14(no. 5-6): 877 - 905, Dec. 2008.

A CD

The content on the CD includes:

1. The scientific article
2. The worksheets
3. MATLAB scripts used during the project
4. MATLAB workspaces containing the results
5. The four standard test images